On the Maintenance of Distributed Storage Systems with Backup Node for Repair

Gokhan Calis and O. Ozan Koyluoglu
Department of Electrical and Computer Engineering
University of Arizona
Tucson, Arizona 85721
Email: {gcalis, ozan}@email.arizona.edu

***Abstract*—We consider a hierarchical DSS where the content is stored with coding in storage nodes and without coding in a backup node. We analyze the system where mobile nodes request the content from the storage nodes. Backup node is assumed to be accessible by storage nodes only in the case where repairs are required. Under this scenario, we derive the upper bound on the file size bound as well as establish critical points on the trade-off curve known as Minimum Storage Regenerating (MSR) and Minimum Bandwidth Regenerating (MBR) points. Next, we propose optimal code constructions by utilizing existing regenerating codes. Furthermore, we analyze the statistics of maintenance and data access costs under Poisson model for node failures and data requests. We derive expressions for expected values and variances of both such costs. Finally, numerical results are provided where we show that the most studied points in the literature (MSR and MBR) are not always optimal with respect to total cost. We also point out that variance of maintenance cost may be much important than variance of data access cost.**

I. Introduction

Evergrowing data and recent interest to analyze and store it brought challenges to how one can maintain data in long term. In particular, to achieve reliable data storage, classical schemes like *replication* are not suitable for today's *big-data* due to their inherent nature of storing the data inefficiently. Henceforth, *coding* techniques are proposed in order to build distributed storage systems (DSS), with improved features, such as increased storage efficiency. One of the most commonly employed method is *Maximum Distance Separable (MDS)* codes. In short, the file to be stored is divided into k fragments and $n-k$ fragments are created from them such that any k fragments are enough for the recovery of the file. Instances of such a coding technique are employed in [1], [2].

Although MDS coding outperforms replication in terms of storage efficiency, recently proposed *regenerating* codes are shown to be improve upon classical MDS codes with data repair efficiencies [3]. Similar to MDS coding, regenerating codes also allow one to recover the file from any k fragments (each of size α). In addition, to regenerate a symbol of the code (i.e., repair a single node), one can download $\beta \leq \alpha$ symbols from any $d \geq k$ nodes. It's shown that such a technique reduces the *repair bandwidth* compared to classical MDS codes [3]. Furthermore, the authors show a trade-off between α and $\gamma = d\beta$ and two ends of the trade-off curve are referred to as *Minimum Storage Regenerating (MSR)* (having minimum α) and *Minimum Bandwidth Regenerating (MBR)* (having minimum γ). Explicit constructions for regenerating codes are further stuided in [4]–[6], and references therein.

In this study, we consider a hierarchical DSS where a dedicated node (referred to as *backup node (BN)*) facilitates the distributed repair process for *storage nodes (SN)*. These storage nodes, in turn, serve the *mobile nodes (MN)*, which can be considered as mobile users asking for the stored data. From MN perspective, there is no BN and BN only plays a role in repair process of SN, where the system operated in a hierarchical manner. Toy example of hierarchical DSS is given in Fig. 1. BN can be useful in scenarios like i) one additional node may increase reliability of the system in terms of mean time to data loss due to increased redundancy in the system which prolongs the time it takes to reach a system state where the file is no longer recoverable, ii) BN can be utilized in wireless caching. Given this setup, we have the following results. We first calculate an upper bound on the file size that can be stored in this hierarchical system. Next, we characterize fundamental limits of this model using this bound and propose optimal code constructions that we build on existing regenerating codes [4]–[6]. Third, we analyze DSS with respect to maintenance and data access cost under Poisson failure and request models and analyze the statistics of these cost measures. With this analysis, we show the trade-off between α and γ translates into a trade-off between maintenance and data access costs. Interestingly, we show that minimum total cost may not be achieved at end points whereas a code operating at intermediate points on the $\alpha - d\beta$ tradeoff can be optimal with respect to this total cost measure. Finally, we show that data access cost may experience very small variance compared to maintenance cost.

Related Work: In [7], device-to-device (D2D) mobile caching network is studied where the objective is to minimize energy consumption of data retrieval by use of replication. Similar to our work, [8] analyzes a D2D system (with coding) where some of the mobile users are utilized as storage nodes and the remaining users may request the file. In addition, the model includes a base station that stores the whole file. In [8], storage nodes and mobile nodes are not distinguished in data retrieval case. That is, eventhough storage nodes are already equipped with some data, they still contact k nodes to access the data. Furthermore, in our model, we assume that BN is always participating in the repair process, whereas in [8], base

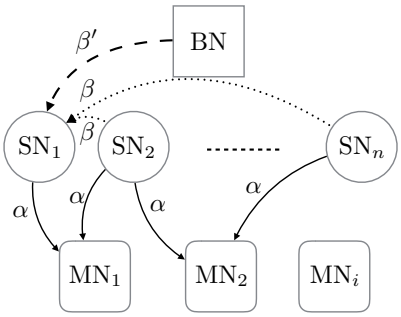

Fig. 1: Hierarchical DSS

station is involved in repairs only when number of storage nodes goes below d.

II. System Model

A. Hierarchical DSS

In this work, we assume there is a hierarchical DSS in which there are three separated set of nodes, a backup node (BN), storage nodes (SN) and mobile nodes (MN) as shown in Fig. 1.

- BN stores information (of size α') regarding file $\mathbf{f}$ of size $\mathcal{M}$. We first consider $\alpha' = \mathcal{M}$ for BN and later reduce this capacity. BN can serve SN such that if failures occur within SN, they can be repaired with the help of BN (and other SN). However, MN cannot access to BN. (This models cellular networks such as femtocells and storage system with backup.)
- There are n SN where each one of SN stores α amount of encoded data. Repairing a failed SN requires to download β amount of data from any d SN as well as downloading $\beta' = \tau\beta$ amount of data from BN.
- MN represents the mobile users who may ask for a file from DSS. MN asking for a file can contact any k SN and download α from each in order to access to the original file $\mathbf{f}$.

In the following section, we analyze $(\alpha, \gamma = d\beta + \beta', \alpha')$ trade-off when $\alpha' \leq \mathcal{M}$, because storing more than the file size is unnecessary. We characterize the trade-off for (α, γ) for $\alpha' = \mathcal{M}$ and then show that same trade-off can be achieved when $\alpha' = \frac{\beta'\alpha}{\beta}$.

B. User Requests and Node Failures

1) User Requests: MN file requests are modeled as a Poisson process with independent and identically distributed (i.i.d.) inter-request time T_r which follows a pdf

$$f_{T_r}(t) = \omega e^{-\omega t}, \quad \omega \geq 0, \quad t \geq 0, \tag{1}$$

where ω is the expected data request rate from MN.

2) Node Failures: Node failures within SN are also modeled as Poisson process with i.i.d. inter-failure time T_f which follows a pdf

$$f_{T_f}(t) = \mu e^{-\mu t}, \quad \mu \geq 0, \quad t \geq 0, \tag{2}$$

where μ is the expected failure rate for one of SN.

C. Communication Cost

We denote the cost for one of the SN to download one symbol from other SN and BN as $\rho_{\text{SN}}^{\text{SN}}$ and $\rho_{\text{SN}}^{\text{BN}}$ respectively. On the other hand, for MN to download one symbol from SN is denoted by $\rho_{\text{MN}}^{\text{SN}}$. Also, we define $\rho \triangleq \frac{\rho_{\text{SN}}^{\text{BN}}}{\rho_{\text{SN}}^{\text{SN}}}$ and it's assumed that $\rho \geq 1$, that is for SN, the cost of downloading from SN is at most same as downloading from BN. (Downloading from BN is more costly.)

D. Data Access and Maintenance

1) Data Access: Consider a code $\mathcal{C}$, which maps $\mathcal{M}$ symbols (over $\mathbb{F}_q$) in $\mathbf{f}$ to length n codewords (nodes) $\mathbf{c} = (c_1, \ldots, c_n)$ with $c_i \in \mathbb{F}_q^{\alpha}$ for $i = 1, \ldots, n$. These codewords are distributed to n SN. We have the following data access property for MN.

Definition 1 (Data Access Property)**.** *The file $\mathbf{f}$, which consist of $\mathcal{M}$ elements in $\mathbb{F}_q$, is encoded to n codeword symbols (each in $\mathbb{F}_q^{\alpha}$) such that $\mathbf{f}$ can be decoded by accessing any k of them.*

2) Maintenance Process: If one of the SN fails, it's required to regenerate the stored data that is lost due to failure so that the file is maintained with the same tolerance as before. Throughout this study, scheduled maintenance is assumed, where the repairs are performed periodically where each repair epoch is Δ seconds. In other words, the time between two consecutive maintenance processes is denoted with $\Delta \geq 0$ and $\Delta = 0$ case is referred to as instantaneous repair, which is often the case for classical regenerating codes discussed in the introduction. We have the following maintenance process definition.

Definition 2 (Maintenance Process)**.** *After Δ amount of time, any failed node can be reconstructed by downloading β symbols from any d SN and β' symbols from BN. If there are at most $d-1$ SN available after Δ, then repairs are performed using BN only and downloading α symbols, if BN stores the entire file.*

As a result of maintenance process described above, each failed node is reconstructed by using both SN and BN communication and downloading $d\beta + \beta'$ amount of data, only if there are d or more SN are available at the time of repair. Otherwise, each repair is performed using BN only, in which a regenerated node downloads α amount of data directly from BN. Henceforth, cost of repair of a node is either $\rho_{\text{SN}}^{\text{SN}} d\beta + \rho_{\text{SN}}^{\text{BN}} \beta'$ or $\rho_{\text{SN}}^{\text{BN}} \alpha$, depending on the number of available (live) storage nodes in the network at the end of each epoch.

Remark 3. *We note that for $\Delta \neq 0$, there exists a positive probability that the system will have less than k (or even 0) number of alive SN at the end of a maintenance epoch (Δ seconds). Therefore, for $\Delta \neq 0$, DSS may experience data loss if $\alpha' < \mathcal{M}$ as SN may not be reconstructed from BN and remaining (live) SN.*

III. File Size Bounds

In this section, we perform an analysis to obtain file size bounds for hierarchical DSS. In order to keep storage system

functional, n SN (each storing α) needs to be maintained. If a node fails, a newcomer node needs to be regenerated. The newcomer node downloads β symbols from any d SN as well as $\beta' = \tau\beta$ from BN as mentioned in the previous section. The resulting repair bandwidth is $\gamma = d\beta + \beta'$ (symbols). In the following, we perform an analysis to find the upper bound on the file size that can be stored in hierarchical DSS.

Theorem 4. *We can bound the file size that can be stored in hierarchical DSS as*

$$\mathcal{M} \leq \sum_{i=0}^{k-1} min\left\{\alpha, (d-i)\beta + \beta'\right\}. \tag{3}$$

Proof. Consider MN connecting k nodes (denoted by an ordered set $\mathcal{O}$ where $\mathcal{O} \triangleq \{1, 2, \ldots, k\}$). Data stored at each node in $\mathcal{O}$ is denoted by $\mathcal{X}_i$ and downloaded data to this node is denoted by $\mathcal{R}_i$. Due to the data reconstruction property, we have $H(\mathbf{f}|\mathcal{X}_\mathcal{O}) = 0$. Accordingly, we have

$$\mathcal{M} = H(\mathbf{f}) = H(\mathbf{f}) - H(\mathbf{f}|\mathcal{X}_\mathcal{O}) = I(\mathbf{f}; \mathcal{X}_\mathcal{O}) \leq H(\mathcal{X}_\mathcal{O}).$$

At this point, we can analyze the bound on the term $H(\mathcal{X}_\mathcal{O})$ in order to obtain file size bound on $\mathcal{M}$. Denote by $\mathcal{O}(i)$ the i^{th} node in the ordered. We can calculate the entropy as

$$\begin{aligned} H(\mathcal{X}_\mathcal{O}) &= \sum_{i=0}^{k-1} H(\mathcal{X}_{\mathcal{O}(i)}|\mathcal{X}_{\mathcal{O}(1)}, \ldots, \mathcal{X}_{\mathcal{O}(i-1)}) \\ &\leq \sum_{i=0}^{k-1} min\left\{\alpha, (d-i)\beta + \beta'\right\}, \end{aligned} \tag{4}$$

which concludes the proof. □

Corollary 5. *Corresponding MSR and MBR bounds can be found as follows.*

$$\alpha^H_{MSR} = \frac{\mathcal{M}}{k}, \quad \gamma^H_{MSR} \geq \frac{\mathcal{M}(d+\tau)}{k(d-k+1+\tau)} \tag{5}$$

$$\alpha^H_{MBR} = \gamma^H_{MBR} \geq \frac{2\mathcal{M}(d+\tau)}{k(2d-k+1+2\tau)} \tag{6}$$

where superscript H is used to denote the hierarchical DSS.

Proof. For MSR bound, we set $\alpha = \frac{\mathcal{M}}{k}$ in (3) and obtain that $(d-i)\beta + \beta' \geq \alpha, \forall i \in [0, k-1]$. Next, observing that $\beta' = \tau\beta$, minimum γ occurs at $\gamma^{\text{H}}_{\text{MSR}} \geq \frac{\mathcal{M}(d+\tau)}{k(d-k+1+\tau)}$ since β is bounded by $\frac{\mathcal{M}}{k(d-k+1+\tau)}$. MBR bound on the other hand follows $\mathcal{M} \leq \sum_{i=0}^{k-1}(d-i)\beta + \beta'$ since $\alpha = d\beta + \beta'$ is the minimum bandwidth. Therefore, β is bounded by $\frac{2\mathcal{M}}{k(2d-k+1+2\tau)}$ and $\gamma^{\text{H}}_{\text{MBR}} = \alpha^{\text{H}}_{\text{MBR}} \geq d\beta + \beta' = (d+\tau)\beta$. □

IV. Optimal Code Constructions

We utilize the existing regenerating codes to obtain optimal codes for hierarchical DSS.

Construction I: Consider a file $\mathbf{f}$ of size $\mathcal{M}$.

- Encode $\mathbf{f}$ using an $[n, k, \tilde{d} = d + \tau]$ MSR/MBR regenerating code.
- Store n encoded symbols in n of SN, where each SN gets one of the symbols. Store the whole file in BN.

Call the output of above construction, $\mathcal{C}^{\text{H}}$(MSR/MBR).

Corollary 6. *Let $\mathcal{C}^H$(MSR/MBR) the code obtained from Construction I. $\mathcal{C}^H$(MSR/MBR) is optimal with respect to Corollary 5 for $\alpha' = \mathcal{M}$.*

Proof. First, assume $[n, k, \tilde{d} = d + \tau]$ MSR code is used to encode file $\mathbf{f}$. Then, the resulting α and β for underlying regenerating code is

$$(\alpha_{\text{MSR}}, \beta_{\text{MSR}}) = \Big(\frac{\mathcal{M}}{k}, \frac{\mathcal{M}}{k(\tilde{d}-k+1)}\Big).$$

During the repair process, a newcomer then downloads $\frac{\mathcal{M}d}{k(\tilde{d}-k+1)}$ from SN. Additionally, it downloads $\tau\beta = \frac{\mathcal{M}\tau}{k(\tilde{d}-k+1)}$ from BN. Since BN stores the whole file, it can compute $\tau\beta$ from it's data. Hence, in total, the repair bandwidth is $d\beta + \tau\beta = \frac{\mathcal{M}(d+\tau)}{k(\tilde{d}-k+1)} = \frac{\mathcal{M}(d+\tau)}{k(d-k+1+\tau)}$. Therefore, we can achieve $(\alpha^{\text{H}}_{\text{MSR}}, \gamma^{\text{H}}_{\text{MSR}})$ in Corollary 5.

On the other hand, consider $[n, k, \tilde{d} = d + \tau]$ MBR code is used to encode file $\mathbf{f}$. Then, the resulting α and β for underlying regenerating code is

$$(\alpha_{\text{MBR}}, \beta_{\text{MBR}}) = \Big(\frac{2\mathcal{M}\tilde{d}}{k(2\tilde{d}-k+1)}, \frac{2\mathcal{M}}{k(2\tilde{d}-k+1)}\Big).$$

A newcomer downloads $\frac{2\mathcal{M}d}{k(2\tilde{d}-k+1)}$ from the SN as well as it downloads $\tau\beta = \frac{2\mathcal{M}\tau}{k(2\tilde{d}-k+1)}$ from BN. BN can compute $\tau\beta$ since it stores the whole file. Therefore, total repair bandwidth is $d\beta + \tau\beta = \frac{2\mathcal{M}(d+\tau)}{k(2\tilde{d}-k+1)} = \frac{2\mathcal{M}(d+\tau)}{k(2d-k+1+2\tau)}$. Hence, $(\alpha^{\text{H}}_{\text{MBR}}, \gamma^{\text{H}}_{\text{MBR}})$ that is computed in Corollary 5 can be achieved using regular MBR codes. □

Remark 7. *If $\tau < k$, instead of storing whole file of size $\mathcal{M}$ in BN, we can improve the rate of the code by storing only $\tau\alpha$ amount of data in BN. Formally, instead of encoding $\mathbf{f}$ with $[n, k, \tilde{d} = d + \tau]$ regenerating code, we can use $[\tilde{n} = n + \tau, k, \tilde{d} = d + \tau]$ regenerating code. Here, any n out of $\tilde{n}$ encoded symbols are distributed to n SN and the remaining τ symbols are stored as a super-node in BN. The resulting code is still optimal and achieves MSR and MBR bounds that are found in Corollary 5. For the same k and $\tilde{d}$, both codes have the same α and the first construction stores total of $\mathcal{M} + n\alpha$ amount of data whereas the modified construction stores $\tilde{n}\alpha = (n+\tau)\alpha$. Since $\tau < k$, $\tau\alpha < \mathcal{M}$ and rate is improved.*

Proof. Proof is similar to proof of Corollary 6. Instead of computing $\tau\beta$ from the whole file, BN sends β amount of data from each of τ nodes it stores. □

Remark 8. *In terms of error correction capabilities, if $\tau \geq k$, the system is able to recover from any number of node failures within MN since the BN stores the whole file. However, for the case of $\tau < k$, since BN stores the data which corresponds to τ nodes of regenerating codes, the error correction capability of the system is same as $[\tilde{n} = n+\tau, k, \tilde{d} = d+\tau]$ regenerating code, that is up to any $n - d$ node failures within MN can be tolerated.*

V. Maintenance and Data Access Costs

As discussed earlier, costs of downloading from BN, SN and MN may differ. In this section, maintenance and data access costs are discussed under the Poisson models introduced earlier for scheduled maintenance Δ and $\alpha' = \mathcal{M}$. We denote by $b_i^{(n,p)}$ PMF of binomial distribution with parameters n and p,

$$b_i^{(n,p)} \triangleq \binom{n}{i} p^i (1-p)^{n-i}. \tag{7}$$

A. Maintenance Cost

Under maintenance process discussed earlier, repairs are performed using either both SN and BN (when there are at least d nodes remain in the network) or BN (number of surviving nodes is less than d) only. Accordingly, we can denote the number of nodes that are repaired using BN only and both SN and BN as m_r^{BN} and m_r^{SN} respectively. Denoting the random variable for normalized repair cost per time by C_r, we have the following,

$$\text{E}[C_r] = \frac{\rho_{\text{SN}}^{\text{BN}} \gamma_{\text{BN}} m_r^{\text{BN}} + \rho_{\text{SN}}^{\text{SN}} \gamma_{\text{SN}} m_r^{\text{SN}} + \rho_{\text{SN}}^{\text{BN}} \gamma_{\text{BN}}^{\text{SN}} m_r^{\text{SN}}}{\mathcal{M}\Delta},$$

where we denote the amount of downloads during a node repair from BN only as γ_{BN} (when there are at most $d-1$ surviving nodes), from SN as γ_{SN} and from BN $\gamma_{\text{BN}}^{\text{SN}}$ respectively (when there are at least d surviving nodes).

Theorem 9. *For a DSS considered in previous section with departure rate μ and repair interval Δ, the average repair cost is given by,*

$$E[C_r] = \frac{\rho_{SN}^{SN}}{\mathcal{M}\Delta}\Big(\rho\alpha \sum_{i=0}^{d-1}(n-i)b_i^{(n,p)} + \beta(d+\rho\tau)\sum_{i=d}^{n}(n-i)b_i^{(n,p)}\Big) \tag{8}$$

Proof. Probability that a node has not failed the network during Δ is $p = Pr(T_f > \Delta) = e^{-\mu\Delta}$. Hence, probability that i storage nodes are available for repair is $b_i^{(n,p)}$. For any i, $n-i$ nodes need to be repaired using both SN and BN or BN only. To do BN only repairs, there should be less than d nodes, hence $m_r^{\text{BN}} = \sum_{i=0}^{d-1}(n-i)b_i^{(n,p)}$. Similarly, we have $m_r^{\text{SN}} = \sum_{i=d}^{n}(n-i)b_i^{(n,p)}$ since SN repairs requires at least d live nodes. □

B. Data Access Cost

Denote by p_{SN} the probability that a request for a file is served utilizing SN (in which $k\alpha$ is downloaded from k live nodes). Then, we have C_d as the random variable for data access cost (normalized with file size) and its expected value as

$$\text{E}[C_d] = \frac{\omega}{\mathcal{M}} \rho_{\text{MN}}^{\text{SN}} k\alpha p_{\text{SN}}.$$

Theorem 10. *For a DSS considered in previous section with failure rate μ, request rate ω and repair interval Δ, p_{SN} is given by,*

$$p_{SN} = \frac{1}{\Delta}\sum_{i=k}^{n}\frac{1-p_i}{\mu_i}\prod_{\substack{j=k \\ j\neq i}}^{n}\frac{j}{j-i}, \tag{9}$$

where $\mu_i = i\mu$ and $p_i = e^{-\mu_i\Delta}$.

Proof. See the proof of Theorem 2 in [8]. □

Remark 11. *If there are less than k SN available at the time of file request, DSS would not be able to serve the MN, which happens with probability $1-p_{SN}$. Since MN access is assumed to be restricted to SN, the only solution is to wait for repairs of SN to be over.*

C. Variance Analysis of Costs

In addition to the expected values of costs, second order statistics can also play a major role in practice.

Theorem 12. *Variance of maintenance cost is given by*

$$\begin{aligned} Var(C_r) = \frac{1}{\mathcal{M}^2\Delta^2}\Big((\rho_{SN}^{BN})^2\gamma_{BN}^2\Sigma_{0,d-1} + (\rho_{SN}^{SN}\gamma_{SN} + \rho_{SN}^{BN}\gamma_{BN}^{SN})^2\Sigma_{d,n} \\ - 2\rho_{SN}^{BN}\gamma_{BN}(\rho_{SN}^{SN}\gamma_{SN} + \rho_{SN}^{BN}\gamma_{BN}^{SN})\Sigma_{0,d-1}^{d,n}\Big) \end{aligned} \tag{10}$$

where $\Sigma_{k,j} = \sum_{i=k}^{j}(n-i)^2 b_i^{(n,p)} - (\sum_{i=k}^{j}(n-i)b_i^{(n,p)})^2$ and $\Sigma_{k,j}^{u,v} = \sum_{i=k}^{j}(n-i)b_i^{(n,p)}\sum_{i=u}^{v}(n-i)b_i^{(n,p)}$.

Proof. Due to space limitations, the proof is provided in our technical report [9]. □

Theorem 13. *Variance of data access cost is given by*

$$Var(C_d) = \frac{\omega^2(\rho_{MN}^{SN})^2k^2\alpha^2(p_{SN} - p_{SN}^2)}{\mathcal{M}^2}. \tag{11}$$

Proof. Due to space limitations, the proof is provided in our technical report [9]. □

VI. Numerical Results

In Fig. 2, we analyze our findings under a scenario where $\rho_{\text{SN}}^{\text{SN}} = \rho_{\text{MN}}^{\text{SN}}$ and $\rho_{\text{SN}}^{\text{BN}} = 5$. First, in Fig. 2(a), we show the trade-off between α and γ (trade-off for the per-node storage (bytes) and repair bandwidth (bytes)) as τ changes, where two ends of the trade-off curves are respective MSR and MBR points. As it can be observed, as τ increases, γ reduces for a constant α. The trade-off between α and γ results in a trade-off between C_d and C_r, which is depicted on Fig. 2(b). Since downloading from BN costs much more than SN, increasing τ has a negative impact on C_r.

Interestingly, for the total cost, which is defined as $C = C_r + C_d$, we show that neither end-points of the trade-off curves of Fig. 2(a) are optimal, instead one can achieve a lower C with intermediate points as shown in Fig. 2(c). Lastly, we show how impactful variance is in Fig. 2(b) by analyzing the variances on the trade-off curve. In this set-up, variance of C_d is very small compared to that of C_r, this is because p_{SN} is very close to 1.

We also plot the our findings by changing the value of k in Fig. 2. For $k = 6$, we show the trade-off between α and γ in Fig. 2(d). In Fig. 2(e), it can be observed that increasing k also increased the variance of C_d as the error bars are longer in this case as opposed to error bars on C_d in Fig. 2(b). Finally, we show the trade-off between total cost C and C_d in Fig. 2(f)

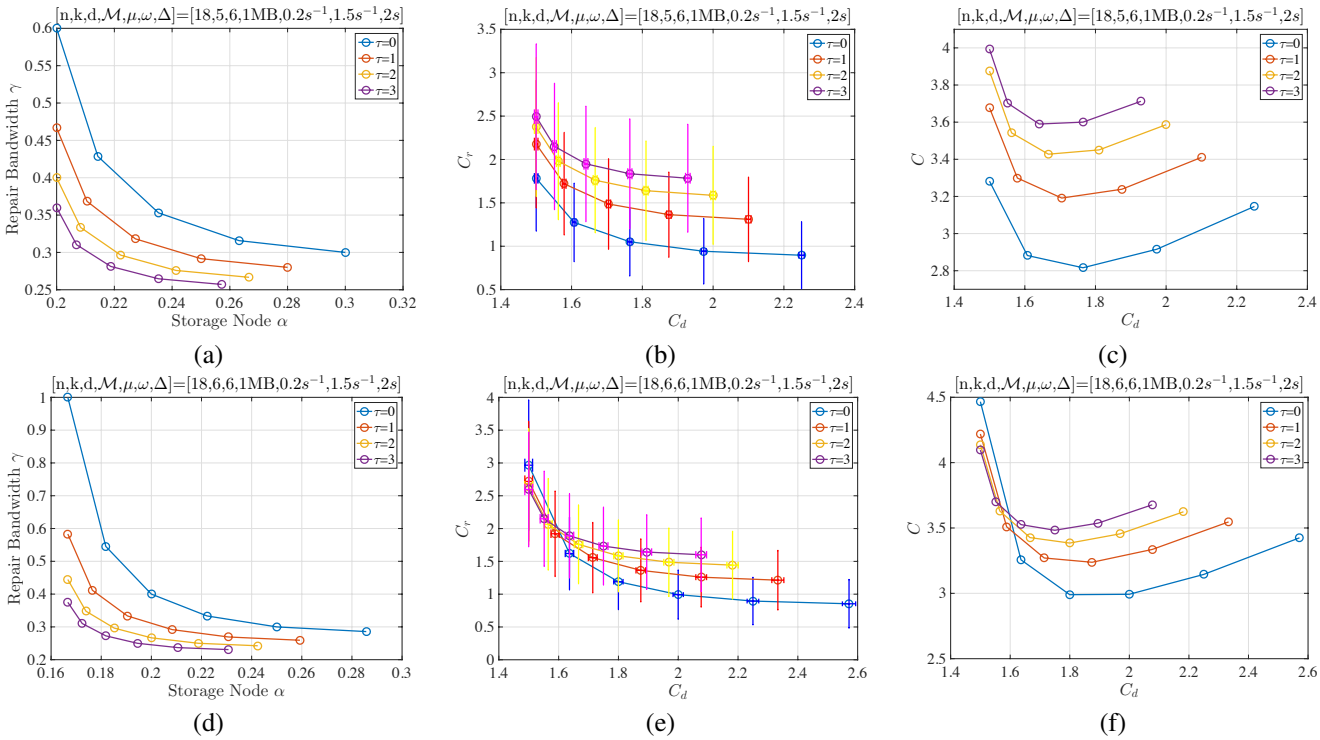

Fig. 2: (a) and (d) Trade-off between α and γ, (b) and (e) trade-off between C_d and C_r and variance of C_r and C_d added as error bars on the curve, (c) and (f) trade-off between C_d and $C = C_d + C_r$

and in this case, the curves are not nicely separated (as they did in Fig. 2(c)). Also, we do not have a specific τ value performing better at all times as different τ values yields better performance for different C_d values.

VII. Conclusion and Future Work

In this work, we studied a hierarchical DSS where BN, SN and MN are separated. Although BN can be accessed by SN, MN can only contact SN. During the repairs of SN, BN can be utilized. It's shown that trade-off between per node storage α and repair bandwidth γ can be obtained for hierarchical DSS. Accordingly, MSR and MBR bounds are established, optimal code constructions (using regenerating codes) achieving those bounds are proposed. In addition, maintenance and data access costs are analyzed using a Poisson model for node failures within SN and file requests of MN. Both expected values and variances are considered in the analysis. Lastly, numerical results on the trade-off curves are provided. Trade-off between maintenance and data access costs is given and it's shown that to achieve better performance in terms of total costs, one may want to operate at intermediate points as well. According to the studied case, it's shown that variance of data access cost may be negligible compared to variance of maintenance cost.

In this study, we analyzed the trade-off for the case when $\alpha' = \mathcal{M}$. In addition, we showed that same trade-off can be achieved for $\alpha' = \tau\beta$ and it requires only a slight modification to the code construction proposed. The trade-off when $\alpha' < \tau\beta$ remains open for now. Furthermore, we discussed the case ($\alpha' < \mathcal{M}$) when DSS may experience data loss. Mean time to data loss can be analyzed using Markov chain models.

References

[1] H. Weatherspoon and J. D. Kubiatowicz, "Erasure coding vs. replication: A quantitative comparison," in *Peer-to-Peer Systems*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2002, vol. 2429, pp. 328–337.

[2] B. Calder, J. Wang, A. Ogus, N. Nilakantan, A. Skjolsvold, S. McKelvie, Y. Xu, S. Srivastav, J. Wu, H. Simitci *et al.*, "Windows azure storage: a highly available cloud storage service with strong consistency," in *Proc. of the Twenty-Third ACM Symposium on Operating Systems Principles*, Cascais, Portugal, Oct. 2011.

[3] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. J. Wainwright, and K. Ramchandran, "Network coding for distributed storage systems," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4539–4551, Sep. 2010.

[4] I. Tamo, Z. Wang, and J. Bruck, "Zigzag codes: MDS array codes with optimal rebuilding," *IEEE Trans. Inf. Theory*, vol. 59, no. 3, pp. 1597–1616, Mar. 2013.

[5] K. V. Rashmi, N. B. Shah, and P. V. Kumar, "Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction," *IEEE Trans. Inf. Theory*, vol. 57, no. 8, pp. 5227–5239, Aug. 2011.

[6] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," *Proc. IEEE*, vol. 99, no. 3, pp. 476–489, Mar. 2011.

[7] J. Pääkkönen, C. Hollanti, and O. Tirkkonen, "Device-to-device data storage for mobile cellular systems," in *Globecom Workshops (GC Wkshps), 2013 IEEE*, Atlanta, GA, Dec 2013.

[8] J. Pedersen, A. G. i. Amat, I. Andriyanova, and F. Brännström, "Distributed storage in mobile wireless networks with device-to-device communication," *CoRR*, vol. abs/1601.00397, Jan. 2016.

[9] G. Calis and O. O. Koyluoglu, "On the maintenance of distributed storage systems with backup node for repair," 2016.